TASK FORCE WEB

Error Handling
9 July 2002

D. Taylor Holmes, P.D.I.T.
Senior Engineer
SPAWAR Systems Center Charleston

Agenda

- Why Error Handling?
- Standard HTTP Errors
- HTTP-Bound UFS Error Handling
  - Example PRI Response Structure
  - PRI Response Elements
- WSDL-Bound UFS Error Handling
  - Example SOAP Fault Structure
  - SOAP Fault Elements

Why Error Handling?

- Graceful Error Handling is an important feature for any software system
- The current NEP environment is a distributed, HTTP-based software system
- The NEP will handle 3 types of errors reported over HTTP(s)
  - Standard HTTP Errors
  - HTTP-Bound UFS Errors
  - WSDL-Bound UFS Errors

Standard HTTP Errors

- The NEP should handle common HTTP Errors returned by a UFS
- The NEP will return a user-friendly HTML page to the Portal Client that corresponds to the specific HTTP Error returned by the UFS
- Common HTTP Errors include:
  - 401, "Unauthorized“
  - 404, “Not Found
  - 500, “Internal Server Error”

HTTP-Bound UFS Error Handling

- HTTP-Bound UFS should use the PRI Response structure to report application specific errors back to the NEP
- The NEP should return a user-friendly HTML page to the Portal Client that displays the error information returned by the UFS in the PRI Response structure
- Show Example of a PRI Response structure returned in the PRIDataResponse Header of the HTTP Response

Example PRI Response Structure

```
<?xml version="1.0"?>
<PRIResponse
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-
   instance"
   xsi:noNamespaceSchemaLocation="http://homep
   ort.navy.mil/schemas/PRIResponse.xsd">
   <ResponseCode>3</ResponseCode>
   <ReturnMessage>Server side error: Could not
   obtain stock quote information for the Ticker
   Symbol XYZ.  Ticker Symbol is not
   valid.</ReturnMessage>
</PRIResponse>
```

PRI Response Elements

- ReturnCode - A numeric value optionally returned by the service module to indicate success or failure of the operation. The following are valid return code values:
  - 0 – Success
  - 1 – Informational
  - 2 – Warning
  - 3 – Fatal
- ReturnMessage – A string optionally returned by the UFS that provides a textual description of the error condition that may have occurred.

WSDL-Bound UFS Error Handling

- WSDL-Bound UFS should use the SOAP Fault structure to report application specific errors back to the NEP
- The NEP should return a user-friendly HTML page to the Portal Client that displays the error information returned in the SOAP Fault structure

Example SOAP Fault Structure

```
<?xml version="1.0"?>
<soap:Envelope
   xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
 <soap:Fault>
   <faultcode>soap:Server</faultcode>
   <faultstring>Ticker Symbol XYZ does not exist</faultstring>

   <faultactor>http://www.myservice.com/services/getstockquote.asp</f
   aultactor>
   <detail>Server side error: Could not obtain stock quote information for
   the Ticker Symbol XYZ.  Ticker Symbol is not valid.</detail>
 </soap:Fault>
</soap:Body>
</soap:Envelope>
```

SOAP Fault Elements

- faultcode - value that identifies the error condition (machine use). SOAP 1.1 defines 4 standard faultcodes (VersionMismatch, MustUnderstand, Client, Server).
- faultstring – string that briefly identifies the fault (meant for user).
- faultactor – the URI that identifies the endpoint where the fault originated
- detail – detailed error information